Laboratorium 4. Term Frequency and Inverse Document Frequency

Dzisiejsze laboratorium skupia się na wykorzystaniu wektoryzacji TF-IDF. Pierwsze zadanie to rozgrzewka – powtórka zadania klasyfikacji z poprzedniego laboratorium, ale z paroma zmianami. Następnie zadanie porównywania dokumentów z użyciem dwóch znanych sposobów. Na sam koniec zadanie wymagające najmniej "kodowania", które skupia się na analizie danych oraz wnioskowaniu.

Zadanie 4.1. Klasyfikacja z TF-IDF

1. Wczytaj dane z katalogów `positive` oraz `negative` z poprzedniego laboratorium
2. Dokonaj wektoryzacji z użyciem `TfidfVectorizer`
3. Utwórz tablicę etykiet (300 zer oraz 300 jedynek) typu `ndarray`
4. Dane podziel na 5 foldów z użyciem funkcji `StratifiedKFold`
5. Utwórz tablice na wyniki
6. Dla każdego foldu:
   - Wytrenuj model z użyciem klasyfikatora `MLPClassifier` na danych **treningowych** (funkcja `fit()`)
   - Dokonaj predykcji modelu na danych **testowych** (funkcja `predict()`)
   - Zapisz w tablicy z punktu 5 dokładność wyrażoną za pomocą metryki `accuracy_score`
7. Wyznacz średnią dokładność oraz odchylenie standardowe i wyświetl wyniki

Dokumentacja
- `TfidfVectorizer`[1]
- `StratifiedKFold`[2]
- `MLPClassifier`[3]

Zadanie 4.2. Podobieństwo w poezji

1. Wczytaj wszystkie dokumenty z katalogu `poezja`
2. Dokonaj wektoryzacji z użyciem `TfidfVectorizer`
3. Wylicz macierz podobieństw cosinusowych z użyciem `cosine_similarity`
4. Wskaż trzy pary najbardziej podobnych do siebie dokumentów (zapisz komentarz w pliku źródłowym)
5. Wyznacz macierz odległości Euclidesowych z użyciem `euclidean_distances`
6. Wskaż trzy pary najbardziej podobnych do siebie dokumentów (zapisz komentarz w pliku źródłowym)
7. Porównaj wyniki z punktu 3 oraz z punktu 5 (zapisz komentarz w pliku źródłowym)

Dokumentacja
- `cosine_similarity`[4]
- `euclidean_distances`[5]

Zadanie 4.3. Częstość wyrazów

Dla poniższych wyrazów wyświetl statystyki TF-IDF na danych z poprzedniego zadania:
- okręt
- wiatr
- fali
- niebo
- się

Czy taka analiza statystyk dla wybranych słów pozwala na pewną kategoryzację dokumentów? Odpowiedź poszerzoną o interpretację uzyskanych wyników, zapisz w postaci komentarza do kodu źródłowego.

[1] <https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html>
[2] <https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html#sklearn.model_selection.StratifiedKFold>
[3] <https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html>
[4] <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html>
[5] <https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html>